(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: 0 196 911
A2

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 86302398.2

(22) Date of filing: 01.04.86

(51) Int. Cl.⁴: H 04 L 11/16
H 04 L 11/08

(30) Priority: 28.03.85 US 716912

(43) Date of publication of application:
08.10.86 Bulletin 86/41

(84) Designated Contracting States:
BE DE FR GB NL

(71) Applicant: HONEYWELL INC.
Honeywell Plaza
Minneapolis Minnesota 55408(US)

(72) Inventor: Kozlik, Tony J.
2936 W. Morrow Drive
Phoenix Arizona 85027(US)

(72) Inventor: Landini, Bruce J.
3031 W. Acoma Drive
Phoenix Arizona 85023(US)

(74) Representative: Geissler, Bernhard, Dr. Patent- und Rechtsanwälte et al,
Bardehle-Pagenberg-Dost- Altenburg-Frohwitter & Partner Postfach 86 06 20
D-8000 München 86(DE)

(54) Local area networks.

(57) In a local area network having a plurality of nodes and utilizing a token passing scheme for communicating between nodes, each node has an interface apparatus which operates independently of a processor associated with said node. The present invention provides for diagnosing and/or recovering a first (remote) node from a second (supervisor or master) node. The first node includes means for monitoring for a message frame addressed to it. If such a message frame is a special function frame, and the node is in an off-line mode, the type of special function frame is determined and the command specified by the type of special function frame is performed. The operation of the first node then returns to the mode of waiting to receive a message frame addressed to it.

EP 0 196 911 A2

Croydon Printing Company Ltd

LOCAL AREA NETWORKS

The present invention relates to distributed data processing systems, particularly of the type known as Local Area Networks (LAN's).

There exists many type of local area networks which are in use, or are under development, or in the process of being standardized. Generally, in a LAN, there exists a bus and a plurality of stations (nodes) attached to the bus. Each station attached to the bus has the same status as any other station, i.e., there is no master and slave relationship between the stations. Thus, no features of a LAN are known to exist whereby a predetermined station can interrogate any other station and further where an interrogation/rely between stations results in a diagnostic and recovery type function.

Accordingly, it is an object of the present invention to provide an apparatus for communicating with a predetermined node of a local area network.

Accordingly the present invention provides a local area network having a plurality of nodes and utilizing a token passing scheme for communicating between nodes, each node having an interface apparatus which operates independently of a processor associated with said node, characterized by means for diagnosing a node from a second (remote) node, comprising: first means for monitoring for a message frame addressed to the first node and determining if the message frame is a special function frame; and second means which, if the message frame is determined to be a special function frame, sets the node to an off-line mode, determines the nature of special function frame and performs the command specified thereby, and returns to the mode of monitoring for a message frame.

An embodiment of the invention will now be described, by way of example, with reference to the drawings in which:

Figure 1 is a system block diagram of a control system in which the present apparatus may be included;

Figure 2 is a schematic block diagram illustrating the common elements of each physical module of the plant control network of the system;

Figure 3 illustrates the waveforms of start-of-frame and end-of-frame delimiters;

Figure 4 shows the format of a token-passing frame;

Figure 5 shows the format of an information frame;

Figure 6 defines the significance of various bit configurations of a destination address field of a frame;

Figure 7 shows a logic block diagram of each bus interface unit of the local area network;

Figure 8 shows the format of a special function frame;

Figure 9 shows the format of a special function acknowledgement frame; and

Figures 10A and 10B, which together comprise Figure 10, are a flow diagram of the operation of the bus interface unit.

INTRODUCTORY SUMMARY

The present system allows a node (remote node) on a local area network to be diagnosed and manipulated from a predetermined node that has an operator interface (or has an interface which permits interaction with an operator outside the local area network). The predetermined node (termed herein a supervisor node) having the operator interface can therefore control, diagnose, and receive status reports from, any of the remote nodes. An interface apparatus is provided at each node of the local area network which permits communication with the predetermined node (supervisor node) independent of the remote node host microprocessor.

The present system comprises an interface apparatus for providing each node of the local area network with the capability of communicating with a supervisor node, independent of the status of the associated node microprocessor. In a local area network having a plurality of nodes, wherein the local area network utilizes a token passing scheme for communicating between nodes and each node has an interface apparatus which operates independently of a processor associated with the node, a method of diagnosing and/or recovering a first (remote) node from a second (supervisor) node is implemented by the first node waiting to receive a message frame addressed to the first node. The

first node then determines if the message frame is a special function frame. If the message frame is determined to be a special function frame, the first node insures that the node is in an off-line mode. The second node then determines the type of special function frame to perform the command specified by the type of special function frame. Finally, the first node operating returns to the step of waiting to receive a message frame addressed to it.

DETAILED DESCRIPTION

Before describing the specific apparatus, it will be helpful to describe a local area netowrk (LAN) in which the apparatus can be utilized. Figure 1 is a system block diagram of a control system 5. The organization, or architecture, of the control system 5 includes a plant control network 14, which is a token pasing distributed local area network (LAN). Physical modules 16 of network 14 are of various specialized functional types, as will be described later. Each physical module 16 is the peer, or equivalent, of the others in terms of right of access to the network's communication medium, the plant control bus 18, for the purpose of transmitting data to other modules 16 of network 14. Highway gateway module 16-HG provides communications and data translation facilities between plant control bus 18 and data highway 20 associated with a process control subsystem 22, which is a distributed digital process control and data acquisition subsystem.

Universal operator station module (US) 16-US of network 14 is a work station for one or more plant operators. It includes an operator console which is the interface between the plant operator or operators and the process or processes of the plant for which they are responsible. Each universal operator station module 16-US is connected to plant control bus 18, and all communications between the universal operator station module 16-US and any other module 16 of network 14 are by means of plant control bus 18. Universal operator station module 16-US has acces to data that is on plant control bus 18 and the resources and data available through or from any of the other modules 16 of

network 14. The universal station modules 16-US includes a cathode ray tube display (CRT) 15 which includes a video display generator, an operator keyboard (KB) 17, a printer (PRT) 19, and can also include (but not shown) a floppy disc data storage device, trend pen recorders, and status displays, for example.

A history module (HM) 16-HM provides mass data storage capability. The history module 16-HM includes at least one conventional disc mass storage device, such as a Winchester disc, which provides a large nonvolatile storage capability. The types of data stored by such a mass storage device are typically trend histories or data from which such trends can be determined, data that constitutes or forms CRT type displays, copies of programs for modules 16, etc.

An application module (AM) 16-AM provides additional data procesing capability in support of the process control functions performed by the controllers associated with the process control subsystem, such as data acquisition, alarming, and batch history collection, and provides continuous control computational facilities when needed. The data processing capability of the application module 16-AM is provided by a processor (not shown) and a memory (not shown) associated with the module.

Computer module (CM) 16-CM uses the standard or common units of all physical modules to permit a medium-to-large scale, general purpose data processing system to communicate with other modules 16 of network 14 and the units of such modules over plant control bus 18 and the units of process control subsystems 22 via the highway gateway module 16-HG. Data processing systems of a computer module 16-CM are used to provide supervisory, optimization, generalized user program preparation and execution of such programs in higher level program languages. Typically, the data processing systems of a computer module 16-CM have the capability of communicating with other such systems by a communication processor and communication lines.

Plant control bus 18 (or more simply bus 18) is a high-speed, bit serial dual redundant communication bus that interconnects all the modules 16 of plant control network 14. Bus 18 provides

the only data transfer path between the principal sources of data, such as highway gateway module 16-HG, application module 16-AM, and history module 16-HM, and principal users of such data, such as universal operator station module 16-US, computer module 16-CM, and application module 16-AM. Bus 18 also provides the communication medium over which large blocks of data, such as memory images, can be moved from one module 16, such as history module 16-HM, to universal station module 16-US. Bus 18 is dual redundant in that it consists of two coaxial cables that permit the serial transmission of binary signals over both.

Each of the physical modules 16 includes certain required standard units, which are illustrated in Figure 2. Each module 16 has a bus interface unit BIU 32 which is connected to the plant control bus 18 by a transceiver 34. Each module 16 is also provided with a module bus 36 which is capable of transmitting 16 bits of data in parallel, a module CPU 38, and a module memory 40. Other units to tailor each type of module 16 to satisfy its functional requirements are connected to module bus 36 so that each such unit can communicate with the other units of a module 16 via its module bus 36. Thus BIU 32 of the module 16 that has the token at any given time is enabled to transmit data on (over) bus 18. All transmissions by a BIU 32 are transmitted in parallel over the coaxial cables which form the bus 18.

Information is transmitted between modules of network 14 by frames of two types, one of which is a token-passing frame (token) 42 illustrated in Figure 4. A token-pass frame 42 includes from 8 to 10 bytes of a preamble 46. Preamble 46 consists of a sequence of logical 1's, and is followed by a start-of-frame delimiter SFD 48 of one byte, a destination address field 50 of two bytes, a source address field 52 of two bytes, a frame check sequence 54 of two bytes which is used to detect errors in frame 42, and an end-of-frame delimiter EFD 56 of one byte. In Figure 3, the waveforms of start-of-frame delimiter SFD 48 and of end-of-frame delimiter EFD 56 are illustrated.

Information transmitted, by a transmit circuit of the BIU 32 of the module 16 having the token, over bus 18 consists of binary

signals which are manchester encoded so that a receive clock can be derived from the received signals by each receiving BIU 32. A logical 0 is transmitted by the signal being low during the first half of a bit and being high during the second half of the bit, a mid-bit low-to-high transition. A logical 1 is transmitted by the signal being high during the first half of the bit and low during the second half, a mid-bit high-to-low transition. Manchester encoding requires that there always be a transition in the middle of each bit cell. If there is no such transition, a code violation (CV) occurs. Both start and end-of-frame delimiters 48, 56 include code violations, four CV's for each. By using CV's in this manner, a 4-bit error would have to occur to change valid data into a frame delimiter. End-of-frame delimiter 56 is used rather than silence on bus 18 because of the possibility of reflections of bus 18 being interpreted as a trasmission after transmission is stopped by the module 16 having the token at any given time. An antijabber timer of each BIU 32 inhibits the continuous transmission of signals by a BIU 32 for more than a predetermined time period which is substantially longer than is required to transmit the largest information frame 44. The timer is reset each time a BIU 32 stops transmitting.

Referring still to Figures 4 and 5, it can be seen that the format of an information frame 44 differs from that of a token-pass frame 42 only by including an information field 58, which is limited to include between 100 and 4088 bytes of binary data. All other fields of an information frame 44 are the same as that of token-pass frame 42.

In Figure 6, the significance of the bit positions of a destination address field 50 of a frame is explained. A frame, either a token-passing frame 42 or an information frame 44, is defined as a packet of a message formatted for transmission over plant control bus 18. There are two basic types of addresses: a physical address and a logical address. A physical address is the address of a given physical module 16, and field 50 is identified or decoded as being that of a physical module 16 when the most significant bit, bit position 15 of field 50, is a 0.

Each module ~~16~~ has a unique 7-bit physical address. Typically, the physical address of a physical module 16 is determined by a multi-bit switch or by a series of mechanically-made circuit interconnections mounted on the printed circuit board of the module containing one of its transceivers 34.

A token-pass frame 42 is identified by the three most significant bits (bits 15, 14, and 13) being 000. The 7 lower order bits, bits 06-00, are the physical address of the physical module 16 to which a token 42, for example, is addressed. If bits 15 to 13 are 001 then the frame is a diagnostic frame addressed to the module 16 whose physical address is given by the 7 lower order bits of field 50. A special function frame is a form of an information frame 44 and can be used to determine if a given problem or set of problems exist in the addressed module. If bits 15 and 14 are 11, then bit 13 designates which of the two receive channels of the addressed module's BIU 32 is to accept and process the frame. If bit 12 is set and the address is a physical address but the frame is not a token or a diagnostic frmae, the frame is addressed to and to be received by all physical modules 16 of network 14.

If bit 15 is 1, the address defined by the lower order bits 12 to 00 is that of a logical entity (logical module), where a logical entity or module is a program module or set of data. If bit 14 is 0 under such circumstances, then bit 13 designates which receive channel of the addressed BIU is to receive, or process, the frame. If bits 15 and 14 are both 1's, then the frame is an interrupt, a high-priority message, and bit 13 designates the receive channel of the BIU 32 to receive the interrupt.

A module's BIU 32 determines which logically addressed frames transmitted over bus 18 are addressed to it. There can be up to 8K (8196) different logical addresses. A BIU 32 determines if it is to accept an information frame with a logical address by means of a logical address filter table. The lower order 10 bits of the address field are the address of bytes of data, a filter byte, which is stored at each addressable location in the filter

memory of each BIU 32. The higher order 3 bits of the logical address, bit positions 12-10, specify a bit position of the addressed filter word. If that bit is a 1, then that frame is accepted by the module since it is addressed to a logical module or program stored in that module's memory 40.

Each of modules 16 includes a module central processor unit 38 and a module memory 40, a random-access memory, and such additional conroller devices (units) as are required to provide the desired functionality of that type of module; i.e., that of the operator station 16-US, for example. The data-processing capabilities of each module's CPU 38 and module memory 40 create a distributed processing environment which provides for improved reliability and performance of network 14 and plant management system 5. The reliability of network 14 and plant management system 5 is improved because, if one module 16 of network 14 fails, the other modules 16 will remain operational. As a result, network 14 as a whole is not disabled by such an occurrence as would be the case in centralized systems. Performance is improved by this distributed environment in that throughput and fast operator response times result from the increased computer processing resources, and the concurrency and parallelism of the data-processing capabilities of the system.

Each module 16 is responsible for the integrity of its own operation. The failure of a module 16 is detected by the module itself and, if it does detect such a failure, it will cease operation and send a terminating state, or status, message if possible. Under certain circumstances, a module's bus interface unit 32 will send out a failed status message if its watchdog timer times out, for example. If a module, as a whole, is unable to send out a failed state or status message, its backup module (not shown) will detect the failure of its primary because of the absence of periodic status messages which are transmitted by a primary module to its backup modules. A backup (secondary) module 16, upon receipt of a terminating or failed status message or upon the absence of the receipt of its associated primary module's status message, starts operating as a primary module.

The functions performed by each physical module are controlled by its programming, and each program entity (program module) is assigned a logical address which is sometimes referred to as a logical module. The backup physical modules will contain the same set of logical functions (logical modules) as their primary. It should be noted that the backup modules for a primary physical module of one type must be of the same type as its primary.

Modules 16 communicate with each other over the communication medium (bus) 18. As mentioned above, in network 14, each of the modules 16 is the equivalent (peer) of the others. Thus, in network 14 no one of the modules 16 is a master module, and each of the modules 16 has an equal right of access for the purpose of transmitting information over bus 14. It should be noted that all modules 16 receive all signals transmitted over bus 14 by any of the other modules. Each module 16 is assigned a physical address, with the smallest physical address of a module 16 being 00 and the largest being 127, so that the maximum number of modules comprising network 14 is 128. While all the modules of network 14 are physically connected to bus 18 so as to both receive and transmit binary data, a logical ring is formed in which each module 16 transmits the token 42 to it successor, the next module 16 in the logical ring having a larger physical address.

A successor module 16 recognizes that it is the successor by transmitting within a predetermined period of time after receiving a token addressed to it by its predecessor, and by doing so accepts the token. Accepting a token addressed to it confers on the accepting module 16 the right to transmit information over bus 18 to other modules 16. Accepting a token requires the accepting module 16 to which the token is addressed to recognize the signals constituting a token as being such, and that the token is addressed to it. The receipt of such a token by a successor module 16 from its predecessor transfers the right to the successor to transmit within a predetermined period of time an information frame over bus 18 to any or all of the modules 16 connected thereto, as well as the necessity for

transmitting the token 42 to its successor. As stated above, a successor module 16 is the module having the next larger physical address than the module having the token at any given time. In any such logical ring, the next larger address after 127 is defined as being logical address 00. It should be noted that, while the maximum number of modules in a logical ring is 128, the minimum number is 2. Each of the modules 16 has a given function, such as being an operator station, a mass-memory storage subsystem, a data processing subsystem, or an access controller which permits other devices including other local-area networks to communicate with network 14, etc.

Figure 7 is a partial functional block diagram of a representative module 16, which includes the bus interface uit BIU 32 and a transceiver 34 which connects BIU 32 to bus 18 and is capable of transmitting data over bus 18 and of receiving data from bus 18. Transceiver 34 is transformer coupled to bus 18. BIU 32 is provided with a very fast microengine 222, one of the functions of which is to identify tokens 42 addressed to it or its module 16 and to transmit a token 42 to its successor module. Microengine 222 is an 8-bit-wide arithmetic and logic unit made of bit slice components, which can execute a 24-bit microinstruction from its programmable read only memory (PROM) 223 in 200 ns, and also includes a crystal controlled clock which produces 5 MHz clock signals.

Data received from bus 18 by BIU 32 is transmitted by bus transceiver 34 and receiver circuitry 224 to receive FIFO register 226, which stores 32 8-bit bytes of data plus 1 parity bit per byte. Microengine 222 examines the destination address fields of data information frames and token pass frames 42 received and stored in FIFO register 226 to determine if each frame received is addressed to it, and, if the frame is addressed to it, whether it is an information frame or a token frame 42. If the received data is an information frame, then the received data is transferred by direct memory access (DMA) write circuitry 228 by conventional direct memory access technique to the module memory 40 over module bus 36 over which the module memory 40 and

module memory CPU 38 directly communicate with BIU 32. Module bus 36 is capable of transmitting 16 data bits plus 2 parity bits in parallel. (Module CPU 38 and module memory 40 are not illustrated here but are shown in Figure 2.)

If a received frame is a token pass frame 42 addressed to BIU 32, i.e., if the token's destination address field 50 contains the address of the BIU (denoted MY ADDRESS), microengine 222 is programmed to act without intervention of the module CPU 40. On recceipt of a token pass frame 42 in which the destination address field 50 of the token is the physical address of module 16 and thus of BIU 32, BIU 32 will transmit an information frame, if one is available, to another module or to all of the modules 16 attached to the bus 18, which form a logical ring. In doing so, microengine 222 causes its DMA read circuitry 232 to transfer data comprising this information from the module memory 40 into its read data FIFO register 234. Microengine 222 causes data from register 234 to be transferred to transmit circuitry 236 8 bits at a time once every 8 instruction cycles (clock periods) of microengine 222. The rate at which data is either obtained from or written into the module memory 40 over module bus 36 by the DMA circuitry 228 or 232 is up to 16 times the rate at which the data is received from bus 18 by buffer receiver register 226 or is transmitted by transmitter circuitry 236 and bus transceiver 34 to bus 18. To assure this is the case, each BIU 32 is assigned the highest priority with respect to direct memory access of the module memory 40.

Module CPU 38 issues commands to BIU 32 by writing the commands into shared registers 238. Microengine 222 processes such commands during interframe gaps or when a frame is being received that is not addressed to it. Shared registers 238 also contain status information that is readable by module CPU 38. BIU 32 is also provided with a random-access memory RAM 240, into which is stored the physical address of module 32 in the network 14, as MY ADDRESS. The source of the signals representing BIU 32's physical address is a series of interconnections on the same circuit board as transceiver 34.

Figure 8 shows a format of a special-function frame. The special function frame can be received by any module 16 of network 14 from any other module 16; however, it is in fact received from a "supervisor" node, specifically the universal station 16-US. As discussed above, the microengine 222 can operate in both an off-line and an on-line mode. In the off-line mode, all commands from the module CPU 38 are processed except the start commands. Also, in the off-line mode, the module 16 does not participate in token passing or receive information frames. The module 16 will process special function frames if they are received. In the on-line mode the module 16 accepts all commands from the module CPU 38 and participates in token passing. The special function frames are not accepted or processed in an on-line mode except for the enter off-line mode frame.

Receipt of special function frames can cause the module 16 to enter the off-line mode, or if it is already in that mode, to perform the function specified by the frame. Special function frames are sent using normal transmission mechanism described above. When a module 16 enters the off-line mode, it disables the module CPU 38 watchdog function, it flags the module CPU 38 that it is in the off-line mode, and it executes the specific diagnostic functions. Special function frames can indicate the following functions:

a) Enter off-line mode.

b) Reset module - performs a reset (hardware master clear) of the module 18.

c) Abort module CPU 38 - the abort is accomplished by pulsing the power fail interrupt line (level 7) and setting appropriate status so that the module CPU 38 enters an externally induced error recovery routine. A module 18 sends a module CPU 38 watchdog time out frame to inform the network 14 that the module CPU 38 in this module has been aborted.

d) One word read - transmits back to the supervisor node the contents of one word of module memory 40 or I/O controller nonpaged status.

e) Return configuration status - transmits back ~~to~~ the supervisor node the module 18 configuration status of BIU 32 which includes a module CPU 38 and module memory 40 status bits.

The reset module special function frame does not have an acknowledgement. All other frames have an acknowledge frame returned, the format of the acknowledgement frame being shown in Figure 9. In the event the module 16 receives a special function frame before it has finished processing a previous special function frame, only the first special function frame will be honoured and have an acknowledgement frame sent.

Figures 10A and 10B together are a flow diagram of the operation performed by the BIU 32, and more specifically the microengine 222 of BIU 32.

The microengine 222 of each module 16 performs a self test (block 101) when the module 16 is initially powered on to verify that the module 16 is operating properly, and starts its interfacing operation at the conclusion of self test at START (block 105). Each BIU 32 of each module 16 receives every frame transmitted on bus 18, and determines if the frame is addressed to it (block 110). If the frame is not addressed to that module, the module 16 is essentially in a wait loop waiting for a frame addressed to the module 16. If the frame received is addressed to the module a check is initiated to determine if the frame is a special function frame (block 115). If the frame is determined to be a special function frame a check is then performed to determine if the frame is a command to enter the off-line mode (block 120). If the command is to enter the off-line mode, the off-line mode flag is set (block 125), a special function acknowledgement frame (or more simply referred to herein as an acknowledgement frame) is queued (block 130), and the operation returns to the wait loop of block 110.

At block 120, if the command was not to enter the off-line mode, a check is performed to determine if the off-line mode has already been established (block 135). If the off-line mode has not been established, the operation returns to the wait loop of

block 110 since special function frame processing can be performed only in an off-line mode. If the off-line mode has been established the command is decoded to determine the type of command, i.e., if a one word read command has been received (block 150). If a one word read command has been received, the word specified is read (block 145), an acknowledgement frame is queued up (block 130) for subsequent transmission when the module 16 receives the token, the acknowledgement frame including the contents of the word specified to be read, and the operation returns to the wait loop of block 110.

At block 140 if the command was not a one word read command, the command is decoded to determine if a return configuration status was received (block 150). If the command is determined to be a return configuration status command, the configuration status is read (block 155), the acknowledgement frame is queued up (block 130), and the operation returns to the wait loop of block 110. If the command was not a return configuration status command (block 150), a determination is made of whether the command is an abort MCPU command (block 160). If the command is to abort the MCPU 38, a level 7 interrupt is issued (block 165), a watchdog time out frame is queued up (block 170), an acknowldgement frame is queued (block 130), and the operation returns to the wait loop of block 110. (The level 7 interrupt is the power fail interrupt which aborts the current processing of the module CPU 38.)

At block 160, if the command is not an abort MCPU command, the command is decoded to determine if a reset module command has been issued (block 175). If the command is not a reset module command, the processing returns to the wait loop of block 110, since none of the established diagnostic command types have been issued. If a reset module command is issued, the microengine 222 issues a reset module instruction (block 180), and waits a predetermined amount of time (block 185) thereby insuring that the reset command has propagated throughout the module and has been effectuated. At the end of the predetermined time period, the operation returns to the start of the operation and performs

the self-test (block 101). It is in this manner, namely by the operation of the microengine 222, that the module 16 can interface with a "supervisor" node, and process the special function frame independent of the operation of the module CPU 38.

After a frame is received and addressed to this module, (block 110) and the frame is determined not to be a special function frame (block 115), a determination is made of whether the frame is a token pass frame (block 282) (Figure 10B). If the frame is a token pass frame a determination is made of whether a special function frame has been queued up (block 284) to be returned to the transmitting module. If a special function frame has been queued, the special function frame is transmitted (block 286), the special function frame being the watchdog timeout frame or a special function acknowledgement frame. After the special function frame has been transmitted, the token is then passed to the successor module 16 (block 288), and the operation returns to the wait loop of block 110. If no special function frame has been queued (block 284), a determination is made of whether the off-line mode has been entered (block 290). If the off-line mode has been entered the operation returns to the wait loop of block 110. If the off-line mode has not been entered, information requested from earlier transmissions/commands is transmitted via an information frame if such information is available (block 292). The token is then passed to the successor module 16 and the operation returns to the wait loop of block 110.

At block 282, if the frame received is not a token pass frame, a determination is made of whether the module is in the off-line mode (block 294). (This branch forms the "normal" transmission between modules apart from the token passing.) At this point the frame must be an information frame since no information can be received in an off-line mode. If the module is in an off-line mode, the operation returns to the wait loop of block 110. If the module is not in an off-line mode, the information frame is received and processed, the processing being performed by the module CPU 38. The commands of the information frame are passed to the module CPU 38 from the microengine 222.

The results of the processing, if any, are queued for subsequent transmission when the module receives the token and is allowed to transmit on the bus 18 (block 296). The operation then returns to the wait loop of block 110.

CLAIMS

1. A local area network having a plurality of nodes and utilizing a token passing scheme for communicating betwen nodes, each node having an interface apparatus which operates independently of a processor associated with said node, characterized by means for diagnosing a first node from a second (remote) node, comprising: first means for monitoring for a message frame addressed to the first node and determining if the message frame is a special function frame; and second means which, if the message frame is determined to be a special function frame, sets the node to an off-line mode, determines the nature of special function frame and performs the command specified thereby, and returns to the mode of monitoring for a message frame.
2. Apparatus according to Claim 1, characterized in that the second means comprises means for:
   a) reading information specified, if the nature of the special function frame is a read command wherein the information to be read is specified;
   b) issuing a control signal to abort the current processing of the processor associated with the node if the nature of the special function frame is an abort command; and
   c) queuing an acknowledgement frame message to be subsequently transmitted to the second (remote) node.
3. Apparatus according to Claim 2, characterized in that the second means further comprises means for:
   a) issuing a reset control signal to the node if the nature of the special function frame is a reset command; and
   b) waiting a predetermined amount of time to allow the reset control signal to propagate through the node.
4. Apparatus according to either of Claims 2 and 3 characterized in that the means for issuing a control signal to abort the current processing comprises means for:

a) issuing a high priority, non-inhibitable interrupt to the processor associated with the node, the interrupt causing the processor to suspend operation; and

b) queuing a time out frame message to be subsequently transmitted to said first node.

5. Apparatus according to any previous claim, characterized in that the first means further comprise means for:

a) determining whether the message frame is a token pass frame, such that if the message frame is not a token pass frame the message frame is processed as a normal transmission message frame, and otherwise proceeding to step (b);

b) determining if a return special function frame is queued, to transmit the return special function frame if it is queued, and then proceeding to step (d);

c) determining if the node is in the off-line mode, to transmit available information frames if the node is not in the off-line mode, and otherwise proceeding to step (e);

d) passing the token to a successor node; and

e) returning to the mode of monitoring for a message frame.

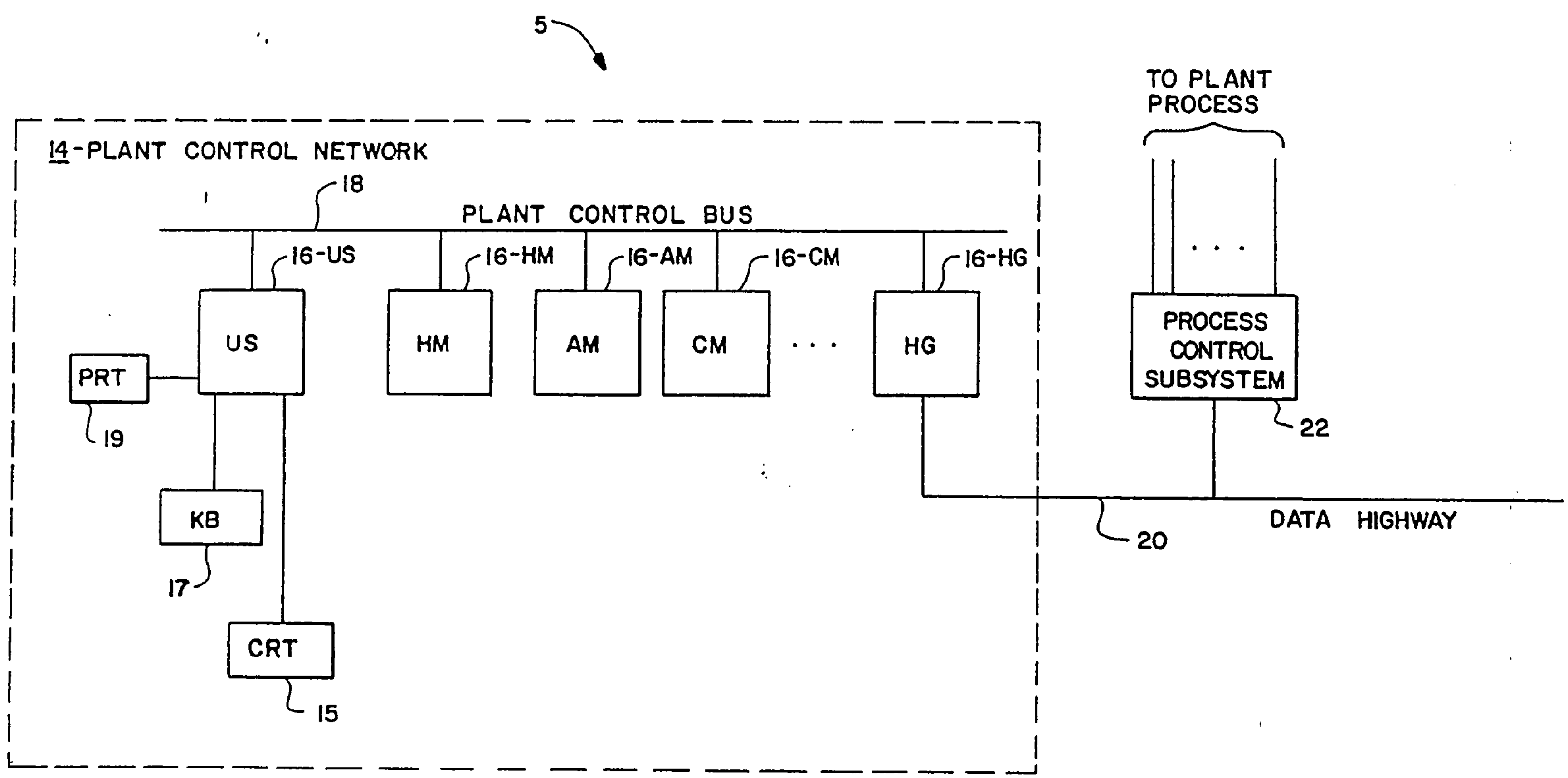

5
14-PLANT CONTROL NETWORK
18
PLANT CONTROL BUS
16-US
16-HM
16-AM
16-CM
16-HG
US
HM
AM
CM
HG
PRT
19
KB
17
CRT
15
TO PLANT PROCESS
PROCESS CONTROL SUBSYSTEM
22
20
DATA HIGHWAY

Fig. 1

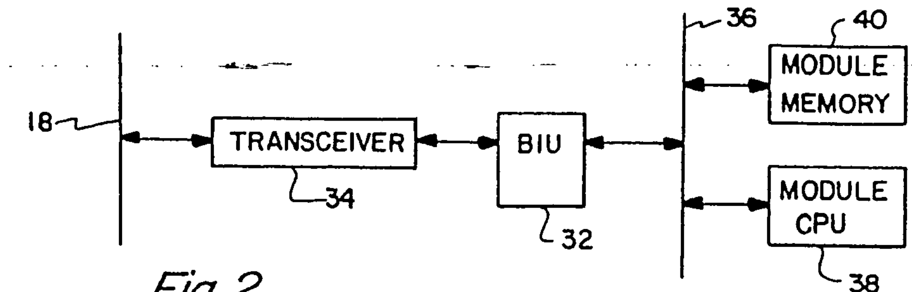


Fig. 2

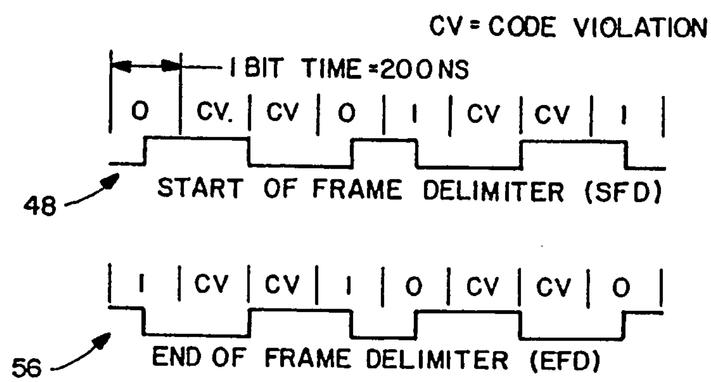


Fig. 3

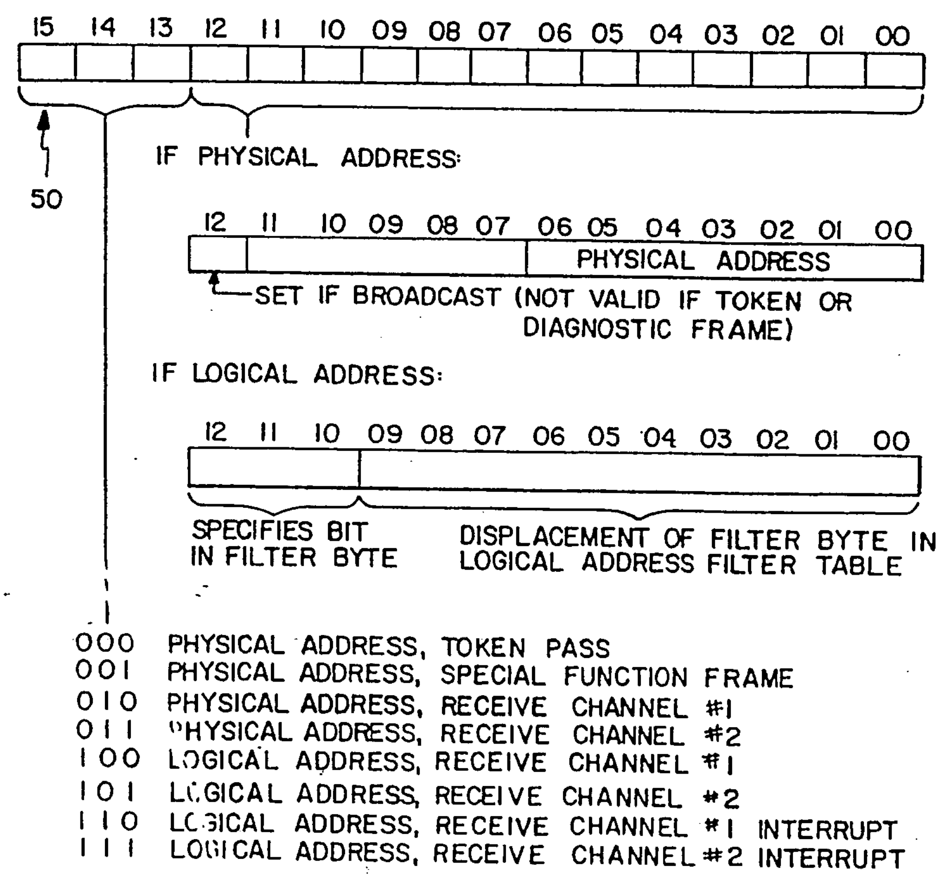


Fig. 6

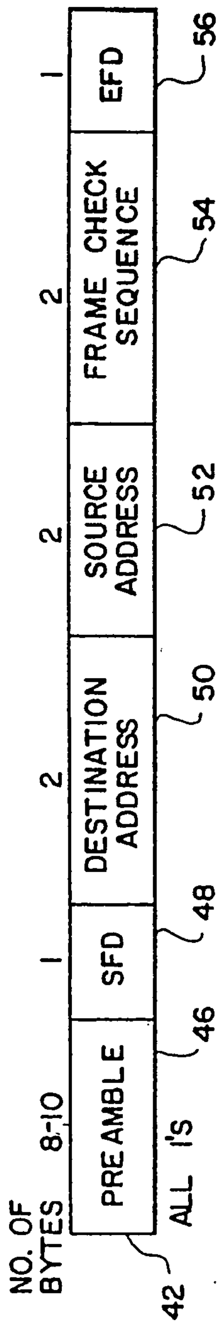

NO. OF BYTES
8-10
1
2
2
2
1
PREAMBLE
SFD
DESTINATION ADDRESS
SOURCE ADDRESS
FRAME CHECK SEQUENCE
EFD
ALL 1'S
42
46
48
50
52
54
56

Fig. 4

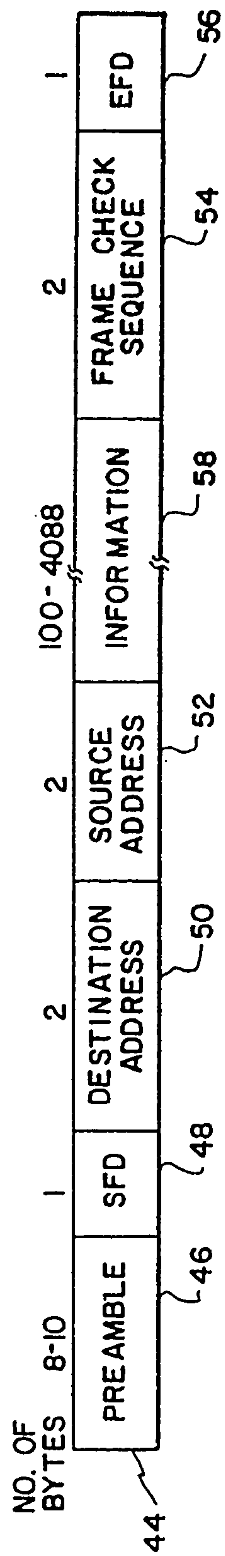
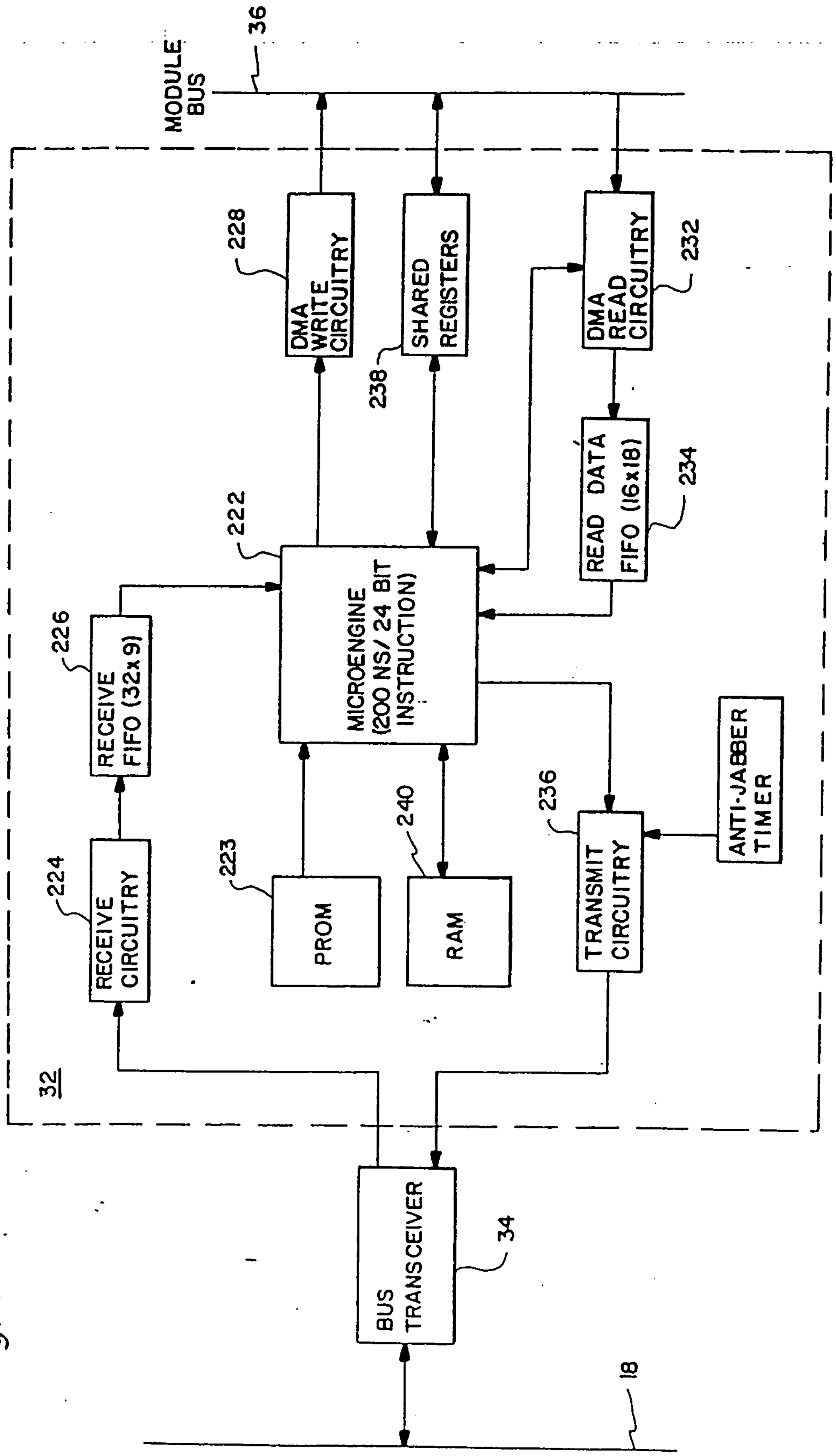

NO. OF BYTES
8-10
1
2
2
100-4088
2
1
PREAMBLE
SFD
DESTINATION ADDRESS
SOURCE ADDRESS
INFORMATION
FRAME CHECK SEQUENCE
EFD
44
46
48
50
52
58
54
56

Fig. 5



MODULE BUS
36
DMA WRITE CIRCUITRY
228
SHARED REGISTERS
238
DMA READ CIRCUITRY
232
READ DATA FIFO (16x18)
234
MICROENGINE (200 NS/ 24 BIT INSTRUCTION)
222
RECEIVE FIFO (32x9)
226
RECEIVE CIRCUITRY
224
PROM
223
RAM
240
TRANSMIT CIRCUITRY
236
ANTI-JABBER TIMER
32
BUS TRANSCEIVER
34
18

Fig. 7

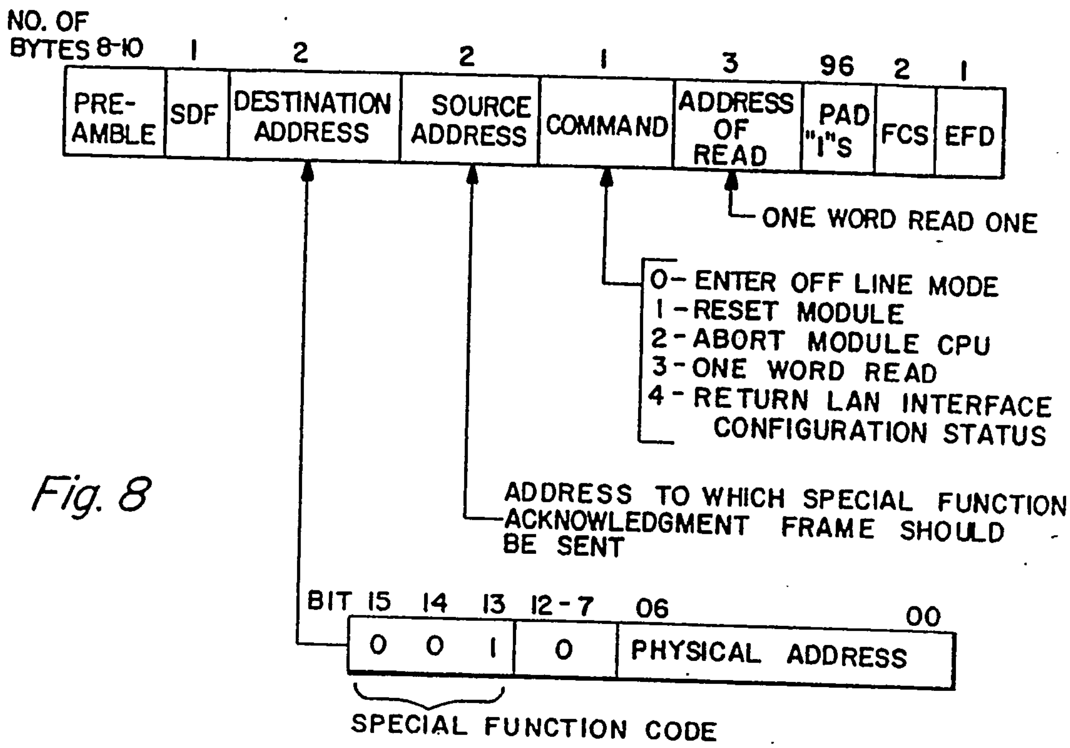

NO. OF BYTES 8-10
1
2
2
1
3
96
2
1
PRE-AMBLE
SDF
DESTINATION ADDRESS
SOURCE ADDRESS
COMMAND
ADDRESS OF READ
PAD "1"S
FCS
EFD
ONE WORD READ ONE
0- ENTER OFF LINE MODE
1 - RESET MODULE
2 - ABORT MODULE CPU
3 - ONE WORD READ
4 - RETURN LAN INTERFACE CONFIGURATION STATUS
ADDRESS TO WHICH SPECIAL FUNCTION ACKNOWLEDGMENT FRAME SHOULD BE SENT
BIT 15 14 13 12-7 06 00
0 0 1
0
PHYSICAL ADDRESS
SPECIAL FUNCTION CODE

Fig. 8

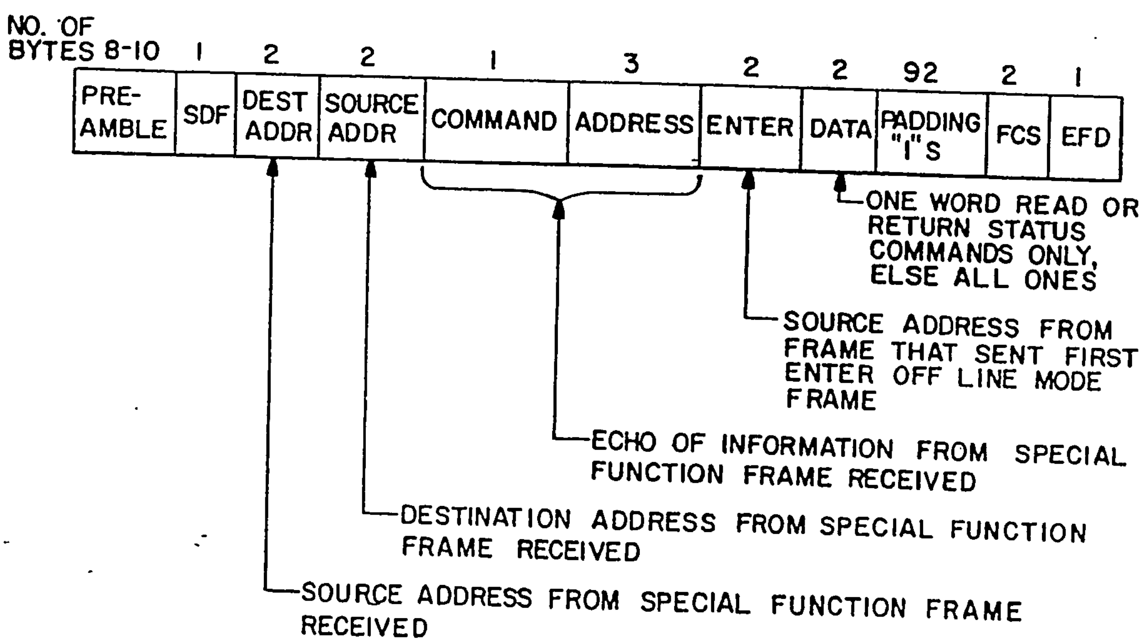

NO. OF BYTES 8-10
1
2
2
1
3
2
2
92
2
1
PRE-AMBLE
SDF
DEST ADDR
SOURCE ADDR
COMMAND
ADDRESS
ENTER
DATA
PADDING "1"S
FCS
EFD
ONE WORD READ OR RETURN STATUS COMMANDS ONLY, ELSE ALL ONES
SOURCE ADDRESS FROM FRAME THAT SENT FIRST ENTER OFF LINE MODE FRAME
ECHO OF INFORMATION FROM SPECIAL FUNCTION FRAME RECEIVED
DESTINATION ADDRESS FROM SPECIAL FUNCTION FRAME RECEIVED
SOURCE ADDRESS FROM SPECIAL FUNCTION FRAME RECEIVED

Fig. 9

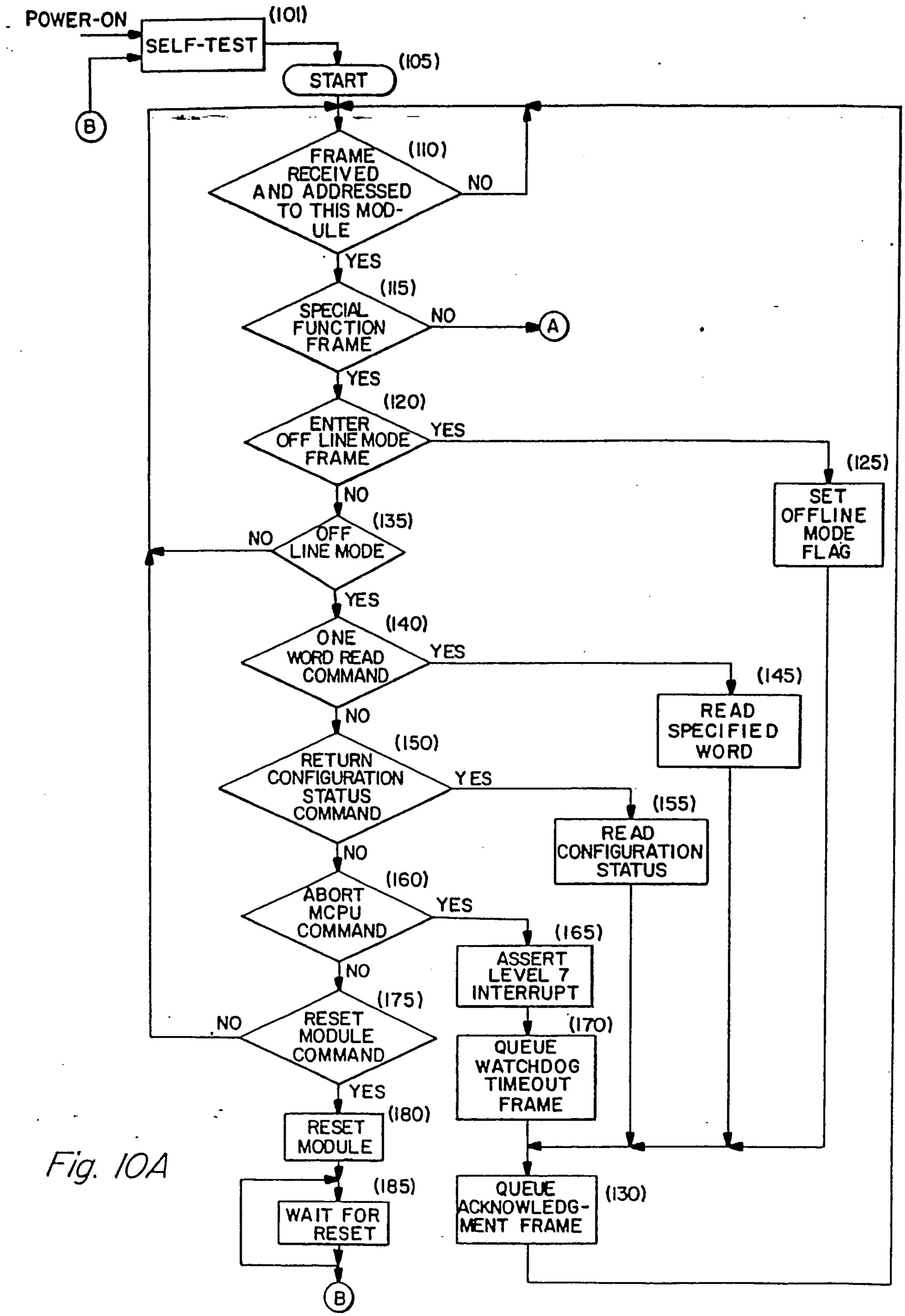

POWER-ON
SELF-TEST
(101)
START
(105)
B
FRAME RECEIVED AND ADDRESSED TO THIS MODULE
(110)
NO
YES
SPECIAL FUNCTION FRAME
(115)
NO
A
YES
ENTER OFF LINE MODE FRAME
(120)
YES
SET OFFLINE MODE FLAG
(125)
NO
OFF LINE MODE
(135)
NO
YES
ONE WORD READ COMMAND
(140)
YES
READ SPECIFIED WORD
(145)
NO
RETURN CONFIGURATION STATUS COMMAND
(150)
YES
READ CONFIGURATION STATUS
(155)
NO
ABORT MCPU COMMAND
(160)
YES
ASSERT LEVEL 7 INTERRUPT
(165)
NO
RESET MODULE COMMAND
(175)
NO
QUEUE WATCHDOG TIMEOUT FRAME
(170)
YES
RESET MODULE
(180)
WAIT FOR RESET
(185)
QUEUE ACKNOWLEDGMENT FRAME
(130)
B

Fig. 10A

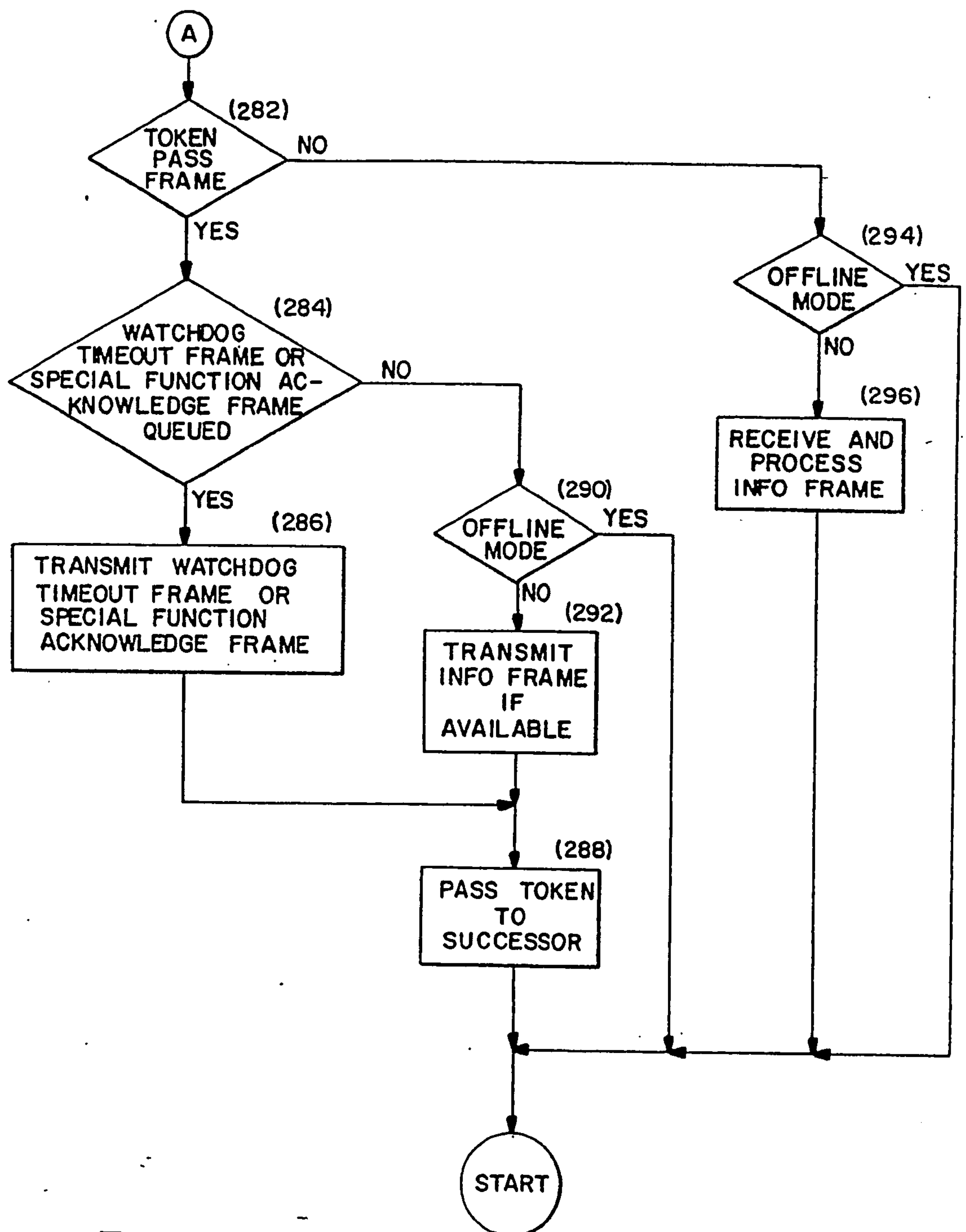

A
(282)
TOKEN PASS FRAME
NO
YES
(284)
WATCHDOG TIMEOUT FRAME OR SPECIAL FUNCTION AC-KNOWLEDGE FRAME QUEUED
NO
YES
(286)
TRANSMIT WATCHDOG TIMEOUT FRAME OR SPECIAL FUNCTION ACKNOWLEDGE FRAME
(290)
OFFLINE MODE
YES
NO
(292)
TRANSMIT INFO FRAME IF AVAILABLE
(288)
PASS TOKEN TO SUCCESSOR
(294)
OFFLINE MODE
YES
NO
(296)
RECEIVE AND PROCESS INFO FRAME
START

Fig. 10B